Training CNNs using high-resolution images of variable shape

Ferran Parés*, Dario Garcia-Gasulla*, Jesús Labarta*†
*Barcelona Supercomputing Center, Barcelona, Spain
†Universitat Politècnica de Catalunya, Barcelona, Spain
E-mail: {ferran.pares, dario.garcia, jesus.labarta}@bsc.es

Keywords—Neural Network, CNN, Medium dataset, High-memory.

I. Extended Abstract

Neural Networks, concretely Convolutional Neural Networks, have proven empirically to be capable of successfully solve image recognition problems. Usually, image recognition datasets are composed by a set of images that have been pre-processed to have exactly the same width and height, since consistency between input shapes is a mandatory condition for typical Neural Network architectures and a really helpful condition for training them successfully. Handling images of varying size and varying aspect ratio during a CNN training process have not been tackled in the literature and presents several challenges like adapting its architecture to avoid the equal image shape limitation, how to group images in batches, how the CNN is going to learn patterns at this wide range of sizes, how to generalize this new unprocessed problem, among others.

Additionally, image recognition images are usually down-sampled to a size of 256x256 or similar, which is a really low resolution that allows to fit in memory all the activations produced in the typical CNN architectures. In fact, using images of greater size arises memory consumption issues and new challenges from the point of view of learning.

So, the main objective of this work is to successfully train a CNN using high resolution images of varying size and aspect ratio.

A. Dataset

In this project we used the Medium dataset, a subset of the recently available Met dataset [1]. This dataset consist of several museum pieces like sculptures, paintings, photographs and its main material they are composed by. As you may expect, the problem is to predict the main material of the museum piece based on its image. This dataset contains images of varying size and aspect ratios, so it is suitable for this project. Smaller images have 0.25MP with shape 500x500 (width x height) and larger images have 16MP with shape 4000x4000 (Figure 1a), while aspect ratios vary from landscape images like 2000x1173 (Figure 1b) to portrait images with 572x1026 dimensions (Figure 1c).

B. CNN architecture

As said before, traditional CNN architectures are unable to work with images of varying shape. That's mainly due to fully connected layers, since these layers require a fixed input shape. In contrast, convolutional layers can work with different input shapes. So, there exists in the literature a specific type of CNN architecture called fully-convolutional that allows to work using variable input shapes. The main characteristic of this architecture is that it aggregates all the spatial information to a single value before feeding fully-connected layers, effectively fixing its shape allowing fully-connected usage.

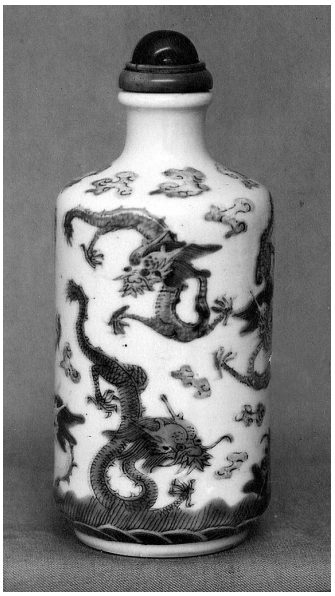
The fully-convolutional layers may solve the ability to feed images of different shapes, but it will be really hard for the CNN to learn patterns at this wide range of scales. For this purpose we propose to use the Spatial Pyramid Pooling (SPP) [2] to handle several pattern scales before discriminating classes.

C. Input pipeline

The architecture proposed in the previous subsection allows to use input images of different shapes on each step, but still needs to join images together in batches. It is possible to join several images of different shapes by padding small images to match biggest ones in the same batch but, it is interesting to

(a) Vase made of terracotta

(b) Spoon made of silver



(c) Vase made of porcelain

Fig. 1: Medium dataset: Subset of images

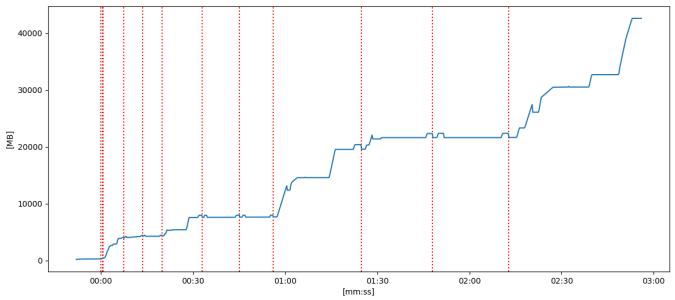


Fig. 2: Memory profile during training process using images from a single bucket. In this case, the training process iteratively doubles the batch size every 3 steps. Initial batch size is 8 and it crashes when trying to use batch size of 64.

group similar images in the same batch to keep in memory as less non-informative padding as possible. Another reason to minimize the padding is the difficulties it generates in terms of learning. It is unknown in the literature how much padding affects training when it gets increased too much.

In order to group images by its shape similarity, we have already designed an input pipeline that we call Buckets to Buckets (B2B). B2B pipeline consists of 2 steps. The first one groups images in superbuckets based on the number of pixels and then, the second one groups images from superbuckets into smaller buckets based on their aspect ratio. However, the performance of this method resides in the right choice of the most appropriate size and aspect ratio boundaries used to separate superbuckets and buckets.

Size and aspect ratio boundaries are chosen based on an heuristic technique. This technique tries to find the set of boundaries that groups images producing as less padding as possible. So, the heuristic orders all the images based on their size or their aspect ratio, places arbitrary boundaries to split them in groups and, finally, iteratively move these boundaries back and forth to reduce the padding of all groups.

D. Dynamic batch size

Since we have grouped images in superbuckets based on its image size, there will be groups with small images and groups with larger ones. So, when using large images the pipeline will batch as much as possible while fitting in memory, but when using smaller ones the pipeline will be capable of batching more images and still fit in memory. So, on each step, the batch size used will vary depending on the bucket images are coming from.

However, the dynamic batch size also brings some drawbacks. From the learning point of view the system has to adapt its learning rate depending on the batch size in order to get training stability [3], and from the machine point of view, the dynamic batch size brings memory usage variability.

E. Memory profiling

Memory profiling has been a great tool in this project to detect memory instability, memory reduction when applying different B2B boundaries, batch size limits, etc. Memory profiles are build from tracking the process resident memory during training and then visualized in plots like the one in Figure 2.

F. Current limitations to overcome

Currently this project uses Tensorflow and Marenostrum4 nodes. The main reason to use Marenostrum4 nodes its the amount of memory that nodes offer (96GB on regular nodes and even more on the high-memory nodes). Regardless of having 96GB available on the MN4 node, we are only capable of using around 40GB of this memory due to a Tensorflow limitation on its implementation. Tensorflow indexes some of its tensors with an int32 variable, so the library is unable to use tensors bigger than 2.147.483.648 positions.

G. Conclusions

This project offers a lot of challenges from the point of view of Artificial Intelligence and High-Performance Computing. There is no standard procedure to train CNN using images of varying size and aspect ratio in the AI literature, while the use of high-resolution images introduces a challenge in terms of memory consumption.

Some of the problems are already tackled and explained in this extended abstract and some of them remain to be solved. The main goal of this project is to successfully train a CNN under these dataset conditions and, train it as fast as possible using greater memory consumption.

References

[1] M. Museum, "Open access initiative," https://github.com/metmuseum/openaccesse, 2017.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[3] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He, "Accurate, large minibatch sgd: Training imagenet in 1 hour," *arXiv preprint arXiv:1706.02677*, 2017.

Ferran Parés received his BSc degree in Electronics in the Universitat Politècnica de Catalunya (UPC), Spain in 2014. Afterward, he completed his MSc degree in Artificial Intelligence in 2016 from a MSc program with three participating universities: Universitat Politècnica de Catalunya (UPC), Universitat de Barcelona (UB) and Universitat Rovira i Virgili (URV). The same year, he joined Barcelona Supercomputing Center (BSC) in the High-Performance Artificial Intelligence group and started his PhD in Artificial Intelligence from Universitat Politècnica de Catalunya (UPC) in the following year (2017), Spain.